

Performing large full-wave simulations by means of a parallel MLFMA implementation

B. Michiels, J. Fostier, I. Bogaert, D. De Zutter
 Department of Information Technology (INTEC)
 Ghent University (UGent)
 Ghent, Belgium
 bart.michiels@intec.ugent.be

Abstract—In this paper large full-wave simulations are performed using a parallel Multilevel Fast Multipole Algorithm (MLFMA) implementation. The data structures of the MLFMA-tree are partitioned according to the so-called hierarchical partitioning scheme, while the radiation patterns are partitioned in a blockwise way. To test the implementation of the algorithm, a full-wave simulation of a canonical example with more than 50 millions of unknowns has been performed.

I. INTRODUCTION

One of the most popular methods to solve electromagnetic scattering problems is the use of boundary integral equations, discretized by means of the Method of Moments (MoM). This method discretizes the scattering problem into a set of N linear equations, with N the number of unknowns on the discretized surface(s) of the scatterer(s). Solving this set of equations iteratively using Krylov-based methods would require $\mathcal{O}(N^2)$ operations for each matrix-vector product. To lower the complexity of the MoM, the Multilevel Fast Multipole Algorithm (MLFMA) is applied. Both for memory and time one obtains a complexity of $\mathcal{O}(N \log N)$, which makes it feasible to perform simulations with a large number of unknowns.

During the past decade the increase of computational power has mainly been achieved by increasing the number of cores in a CPU, rather than improving the performance of an individual CPU-core. Therefore it is important to develop efficient parallel algorithms. Scalability is an important property for parallel algorithms. Consider an algorithm with a serial complexity of $C(N)$ and where the number of CPU-cores (P) grows proportionally with N , so $P = \mathcal{O}(N)$. In that case the parallel algorithm is scalable if the complexity of none of the CPU-cores exceeds $\mathcal{O}(C(N)/P)$. For the MLFMA this means that the complexity of every CPU-core should not exceed $\mathcal{O}(\log N)$. In what follows we discuss such a scalable MLFMA technique.

II. PARTITIONING OF THE MLFMA

In this paper scattering problems are considered for which the discretization is inversely proportional to the frequency and not much smaller than the wavelength, e.g. a $\lambda/10$ -discretization.

In the MLFMA the unknowns are subdivided in a tree-like structure of boxes. At the lowest levels there are $\mathcal{O}(N)$ MLFMA-boxes, containing $\mathcal{O}(1)$ – i.e. independent of N – unknowns. The radiation patterns of the boxes at these levels contain $\mathcal{O}(1)$ sampling points. For each level one goes up in the MLFMA-tree, the number of MLFMA-boxes decreases approximately by a factor of four, while the number of sampling points of the radiation patterns increases roughly by the same factor of four. Consequently, at the top levels there are $\mathcal{O}(1)$ MLFMA-boxes with radiation patterns that have a size of the order N . Each MLFMA-level has a total number of $\mathcal{O}(N)$ sampling points, and, as there are $\mathcal{O}(\log N)$ MLFMA-levels, the total complexity of the MLFMA is $\mathcal{O}(N \log N)$.

As mentioned before, in order to obtain scalability, for each CPU-core the complexity must not exceed $\mathcal{O}(\log N)$ or, equivalently, the complexity per level should be $\mathcal{O}(1)$.

A straightforward way to partition the MLFMA-tree among the CPU-cores is to assign each MLFMA-box, with its whole radiation pattern, to a certain CPU-core. This is called spatial partitioning. At the bottom levels one can subdivide the $\mathcal{O}(N)$ MLFMA-boxes among $\mathcal{O}(N)$ CPU-cores. However this is not possible at the top levels, as there are only $\mathcal{O}(1)$ boxes. The CPU-cores that are attributed an MLFMA-box at a top level contain $\mathcal{O}(N)$ sampling points, violating the condition for scalability.

To solve the complexity problems for spatial partitioning at the top levels, hybrid partitioning was introduced, which

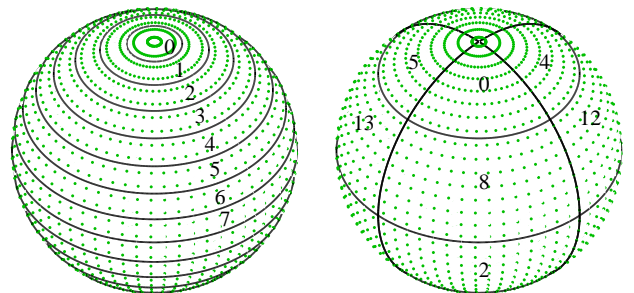


Fig. 1. Stripwise (left) vs. blockwise (right) partitioning of the sampling points of a radiation pattern (green dots).

uses k -space partitioning, i.e. the partitioning of the sampling points of a radiation pattern among the different CPU-cores, at the top-levels. This way the $\mathcal{O}(N)$ sampling points of $\mathcal{O}(1)$ boxes are partitioned among $\mathcal{O}(N)$ CPU-cores. However, this still does not lead to a scalable partitioning as, at the middle levels, there are $\mathcal{O}(\sqrt{N})$ boxes containing $\mathcal{O}(\sqrt{N})$ sampling points. One MLFMA-level before the transition from spatial to k -space, the $\mathcal{O}(\sqrt{N})$ boxes have to be partitioned among $\mathcal{O}(N)$ CPU-cores. This is not possible, so the CPU-cores that are attributed an MLFMA-box at this level contain $\mathcal{O}(\sqrt{N})$ sampling points, again violating the condition for scalability.

Hierarchical partitioning solves the problem at the middle levels by using a gradual transition from spatial to k -space partitioning, which can lead to a scalable partitioning scheme [1], [2].

Another matter that influences the scalability is *how* the radiation pattern is partitioned. Fig. 1 shows two possible ways to partition a radiation pattern: stripwise (left) and blockwise (right). The former partitions only the sampling points of the θ -direction, while for the latter sampling points of both the θ - and ϕ -direction are distributed among the different CPU-cores.

The number of sampling points along the θ - and ϕ -direction are both of the order \sqrt{N} . As the number of CPU-cores grows according to $\mathcal{O}(N)$, the $\mathcal{O}(\sqrt{N})$ θ sampling points cannot be distributed among all the CPU-cores in case of stripwise partitioning and the CPU-cores that are assigned to a stripwise partition contain $\mathcal{O}(\sqrt{N})$ sampling points. These scalability problems do not occur when using blockwise partitioning, and as a result blockwise partitioning is required to obtain a truly scalable algorithm [1], [2].

III. RESULTS

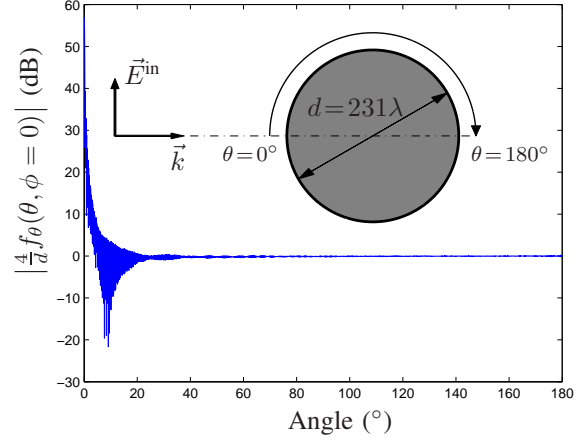
In this section we consider the scattering of a plane wave by a PEC sphere, as depicted in Fig. 2(a). This canonical example is a good test for any MoM-MLFMA-solver, as the results of the simulation can be compared with the analytical solution, i.e. the Mie series.

The sphere, with a diameter d of 231λ , was meshed using a $\lambda/10$ -discretization, resulting in a problem with 50 032 914 unknowns. The MLFMA-tree contained 12 MLFMA-levels and the size of the MLFMA-box at the lowest level was 0.2λ . The simulation was performed in single-precision on a cluster consisting of 8 machines each containing four 8-core AMD Opteron 6136 processors (256 CPU-cores in total) and 64 GByte of RAM (or 2 GByte per core).

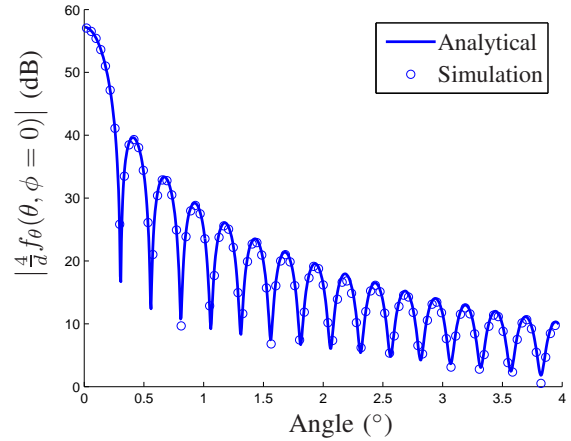
From Fig. 2 one sees the absolute value of $\frac{4}{d}f_\theta(\theta, \phi = 0)$, the θ -component of the normalized radiation pattern in the $\phi = 0$ plane, with d the diameter of the sphere. Fig. 2(a) shows the whole θ -range, sampled in 4544 points, which corresponds to a resolution of 0.04° . Fig. 2(b) depicts the comparison between the result of the simulation and the analytical solution for the forward-scattering direction with θ ranging from 0° to 4° . The error of the radiation pattern is

$$\frac{\|f_\theta(\theta, \phi = 0)_{\text{simulation}} - f_\theta(\theta, \phi = 0)_{\text{analytical}}\|_2}{\|f_\theta(\theta, \phi = 0)_{\text{analytical}}\|_2} = 0.0104$$

which is a typical result for a $\lambda/10$ -discretization [3].



(a) Full θ -range ($0^\circ \dots 180^\circ$) in 4544 sampling points.



(b) Forward-scattering direction ($\theta = 0^\circ \dots 4^\circ$).

Fig. 2. The absolute value of the normalized radiation pattern $\frac{4}{d}f_\theta(\theta, \phi = 0)$ for a PEC sphere with a diameter $d = 231\lambda$.

IV. CONCLUSION

In this paper a parallel MLFMA is implemented using a blockwise hierarchical partitioning scheme. A simulation, involving more than 50 millions of unknowns, of a plane wave impinging on a PEC sphere has been performed on 256 CPU-cores and the results agree with the analytical solution.

ACKNOWLEDGEMENT

The computational resources (Stevin Supercomputer Infrastructure) used in this work were provided by Ghent University.

REFERENCES

- [1] J. Fostier and F. Olyslager, "Provably scalable parallel multilevel fast multipole algorithm", *Electronics Letters*, 44(19):1111-1112, 2008.
- [2] B. Michiels, J. Fostier, J. Peeters, I. Bogaert and D. De Zutter, "Towards an asynchronous, scalable MLFMA for three-dimensional electromagnetic problems", 2011 International Conference on Electromagnetics in Advanced Applications (ICEAA 2011), pp. 66-67, 2011.
- [3] Ö. Ergül and L. Gürel, "Rigorous solutions of electromagnetic problems involving hundreds of millions of unknowns", *IEEE Antennas and Propagation Magazine*, vol. 53, no. 1, pp. 18-27, 2011.